

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 07-175661

(43)Date of publication of application : 14.07.1995

(51)Int.Cl.

G06F 9/445

G06F 13/00

(21)Application number : 05-318688

(71)Applicant : NEC CORP

(22)Date of filing : 17.12.1993

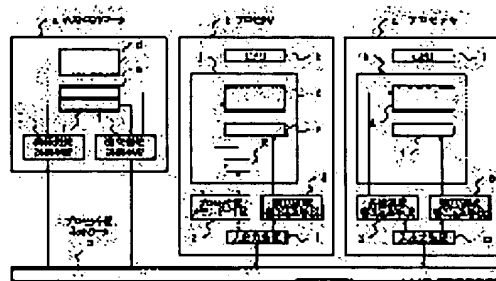
(72)Inventor : SHIMAJIRI TAKUMI

(54) USER PROGRAM LOAD SYSTEM

(57)Abstract:

PURPOSE: To shorten time for loading a user program to plural processors.

CONSTITUTION: When loading an application program existent on a host controller (a) to processors (b) and (c), first of all, a part (d) common for the respective processors is transmitted from the host controller (a) only to the processor (b) by using a common area transmitting means 1. The processor (b), that receives the common part (d), stores the start address and size of the received data in a received data buffer (g) and really writes the data in a local memory (j) by using an inter-processor memory copy means 2. Then, the received data buffer (g) is referred to and each time the total of received data sizes exceeds a fixed value, the common area (d) is copied through an inter-processor network (n) to the processor (c). The processor (c), that receives the common area (d), writes the common area (d) in a local memory (k) by using a common area writing means 3.



LEGAL STATUS

[Date of request for examination] 12.08.1994

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 2906958

[Date of registration] 02.04.1999

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's]

decision of rejection]

[Date of extinction of right]

02.04.2002

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-175661

(43) 公開日 平成7年(1995)7月14日

(51) Int.Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 9/445				
13/00	3 5 1 H	7368-5B		
		9367-5B	G 0 6 F 9/ 06	4 2 0 K

審査請求 有 請求項の数 2 O L (全 4 頁)

(21) 出願番号 特願平5-318688

(22) 出願日 平成5年(1993)12月17日

(71) 出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(72) 発明者 島尻 匠

東京都港区芝五丁目7番1号 日本電気株式会社内

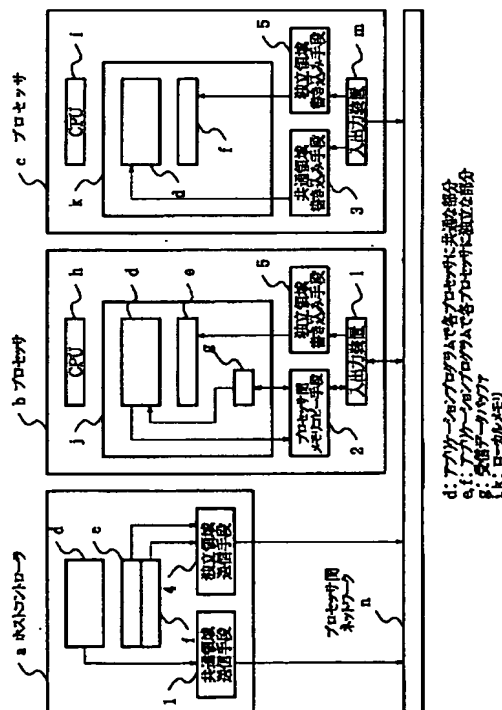
(74) 代理人 弁理士 京本 直樹 (外2名)

(54) 【発明の名称】 ユーザプログラムロード方式

(57) 【要約】

【目的】複数のプロセッサにユーザプログラムをロードする時間を短縮する。

【構成】ホストコントローラ a 上に存在するアプリケーションプログラムをプロセッサ b、c にロードする場合、まず各プロセッサで共通な部分 d についてはプロセッサ b のみへホストコントローラ a から共通領域送信手段 1 を用いて送信する。共通部分 d を受け取ったプロセッサ b はプロセッサ間メモリコピー手段 2 により、受信データバッファ g へ受信したデータの開始アドレスとサイズを格納すると共に、ローカルメモリ j へ実際にデータの書き込みを行う。そして、受信データバッファ g を参照し、受信したデータサイズの合計が一定値を越える毎にプロセッサ c へプロセッサ間ネットワーク n を介して共通な領域 d をコピーする。共通な領域 d を受信したプロセッサ c は共通領域書き込み手段 3 によりローカルメモリ k へ共通な領域 d を書き込む。



【特許請求の範囲】

【請求項 1】 それぞれ独立したローカルメモリ CPU および入出力装置を備えたプロセッサがネットワークで接続され、前記各プロセッサとのファイル入出力処理を行うコントローラを有するマルチプロセッサシステムにおけるプログラムロード方式において、

前記ローカルメモリへ書き込むアプリケーションプログラムをそれぞれのプロセッサで共通な部分と独立した部分に分類しておき、前記共通な部分については 1 つのプロセッサの前記入出力装置のみへ転送する共通領域送信手段と、

前記共通な部分を受信したプロセッサは、指定された前記ローカルメモリへデータを書き込むとともに、その他のプロセッサの入出力装置へ前記ネットワークを介してプロセッサ間でローカルメモリの内容をコピーするプロセッサ間メモリコピー手段と、

前記プロセッサ間メモリコピー手段により前記共通な部分を受信したプロセッサは、当該入出力装置から取り出したデータを指定されるローカルメモリへ書き込む共通領域書き込む手段と、

アプリケーションプログラムで前記各プロセッサで独立な部分については前記コントローラからそれぞれのプロセッサの入出力装置へ前記ネットワークを介して個別に転送する独立領域送信手段と、

前記独立領域送信手段により送り出されたデータを入出力装置から取り出したプロセッサは、それぞれ当該プロセッサのローカルメモリへ書き込みを行う独立領域書き込み手段を設けたことを特徴とする疎結合型マルチプロセッサシステムにおけるユーザプログラムロード方式。

【請求項 2】 前記プロセッサ間メモリコピー手段は、前記受信したデータサイズの合計が一定値を越える毎に機能することを特徴とする請求項 1 記載のユーザプログラムロード方式。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明はユーザプログラムロード方式に関し、特に、各プロセッサが独立したローカルなメモリを持ち、各プロセッサがネットワークで接続され、互いにメモリの内容を複写可能なマルチプロセッサシステムのユーザプログラムのロード方式に関する。

【0002】

【従来の技術】 従来のこの種のユーザプログラムロード方式の一例について、特開平 1-282670 を挙げることができる。その例のブロック図を図 2 に示す。

【0003】 図 2 において、共通データバス 24 にプロセッサ 21～23 が接続する形になっており、ホスト計算機 20 からの制御信号 27 と 28 によりプロセッサへ読み込むデータ切り替えを行っている。

【0004】 実行開始番地保持ブロック 25 はそれぞれのプロセッサ 21～23 に読み込むユーザプログラムの

実行開始番地を保持するための手段である。実行開始番地を実行開始番地保持ブロック 25 に読み込む時には、ホスト計算機 20 からの制御信号 27 をプロセッサ 21～23 対応ごと ON にすることにより、共通データバス 24 から当該実行開始番地保持ブロック 25 へ実行開始番地を取り込み、さらに制御信号 28 を ON にすることでバス選択ブロック 26 を介して、当該プロセッサは実行開始番地保持ブロック 25 からユーザプログラムの実行開始番地を獲得する。そして各プロセッサ 21～23 が上述のようにして実行開始番地を獲得した後、ホスト計算機 20 から全てのプロセッサ 21～23 にユーザプログラムを転送する。

【0005】 プロセッサ 21～23 にホスト計算機 20 からユーザプログラムを転送する時には、制御信号 28 を OFF にすることで共通データバス 24 からバス選択ブロック 26 を介して各プロセッサ 21～23 がデータを読み込むようになっている。したがって、ユーザプログラムを起動する時には、実行開始番地保持ブロック 25 から再度、ユーザプログラム実行開始番地を読み込むことでそれぞれのプロセッサ 21～23 毎に異なる番地にユーザプログラムを読み込み可能な構成になっている。

【0006】

【発明が解決しようとする課題】 上述した従来のユーザプログラムロード方式では、ユーザプログラムをホスト計算機から各プロセッサに対しデータバスを通して個々に転送するので、データバスに接続するプロセッサの数が増えるに従い、転送速度が低下しユーザプログラムのロード時間が長くなるという問題がある。また、ホスト計算機から各プロセッサにロードするアプリケーションプログラムは全く同一のものであることを前提としているので各プロセッサで同一プログラムの引数を変えて別々の条件で動作させることができないという欠点があった。

【0007】

【課題を解決するための手段】 本発明の方式は、それぞれ独立したローカルメモリ CPU および入出力装置を備えたプロセッサがネットワークで接続され、前記各プロセッサとのファイル入出力処理を行うコントローラを有するマルチプロセッサシステムにおけるプログラムロード方式において、前記ローカルメモリへ書き込むアプリケーションプログラムをそれぞれのプロセッサで共通な部分と独立した部分に分類しておき、前記共通な部分については 1 つのプロセッサの前記入出力装置のみへ転送する共通領域送信手段と、前記共通な部分を受信したプロセッサは、指定された前記ローカルメモリへデータを書き込むとともに、その他のプロセッサの入出力装置へ前記ネットワークを介してプロセッサ間でローカルメモリの内容をコピーするプロセッサ間メモリコピー手段と、前記プロセッサ間メモリコピー手段により前記共通

な部分を受信したプロセッサは、当該入出力装置から取り出したデータを指定されるローカルメモリへ書き込む共通領域書き込み手段と、アプリケーションプログラムで前記各プロセッサで独立な部分については前記コントローラからそれぞれのプロセッサの入出力装置へ前記ネットワークを介して個別に転送する独立領域送信手段と、前記独立領域送信手段により送り出されたデータを入出力装置から取り出したプロセッサは、それぞれ当該プロセッサのローカルメモリへ書き込みを行う独立領域書き込み手段を設けたことを特徴とする。

【0008】

【実施例】次に、本発明について図面を参照して詳細に説明する。

【0009】図1は、本発明の一実施例を示すブロック図であり、2台のプロセッサb、cと各プロセッサとのファイル入出力を行うコントローラとしてのホストコントローラaを有するマルチプロセッサシステムである。

【0010】プロセッサb、cにロードするアプリケーションプログラムはホストコントローラa上で各プロセッサで共通な部分と独立な部分に分かれて存在しており、共通な部分をd、独立な部分e、fで示す。各プロセッサで独立な部分e、fについてはプロセッサ台数分必要であり、それぞれプロセッサb、cにロードするためのものとしてホストコントローラa上で分かれて存在している。

【0011】ホストコントローラa上に存在するこのようなアプリケーションプログラムをプロセッサb、cにロードする場合、まず各プロセッサb、cで共通な部分dについてはプロセッサbのみへホストコントローラaから共通領域送信手段1を用いて送信する。共通部分dを受け取ったプロセッサbはプロセッサ間メモリコピー手段2により、受信データバッファgへ受信したデータの開始アドレスとサイズを格納すると共に、ローカルメモリjへ実際にデータの書き込みを行う。

【0012】そして、受信データバッファgを参照し、受信したデータサイズの合計が一定値を越える毎にプロセッサcへプロセッサ間ネットワークnを介して共通な領域dをコピーし、共通な領域dを受信したプロセッサcは共通領域書き込み手段3によりローカルメモリkへ共通な領域dを書き込む。

【0013】また、各プロセッサで独立な部分e、fについてはホストコントローラaから独立領域送信手段4を用いてそれぞれプロセッサb、cへ送信し、ホストコントローラaから受信した独立な領域e、fを受信したプロセッサb、cは独立領域書き込み手段5によりローカルメモリj、kへそれぞれ独立な領域e、fを書き込む。

【0014】アプリケーションプログラムの各プロセッサで共通な部分dには具体的にはアプリケーションプログラムの機械語命令コードと初期化データ、及び非初期

化データが格納されている。データとはアプリケーションプログラムが参照する外部変数、静的変数である。各プロセッサで共通部分dはアプリケーションプログラムをコンパイルして得られるオブジェクトファイルに含まれている。

【0015】また、各プロセッサで独立な部分e、fは、実際にアプリケーションプログラムを実行する時に指定するプログラム引数や実行する時に設定されている環境変数が格納されている。各プロセッサで独立な部分e、fはアプリケーションプログラムをコンパイルした結果であるオブジェクトファイルには含まれず、ホストコントローラaのスタック領域から得られる。

【0016】したがって、各プロセッサで共通な部分dはユーザプログラムをコンパイルした結果のオブジェクトファイルを読み込み、プロセッサbへ転送する。オブジェクトファイルには既に各プロセッサにロードするアドレスが定義されているのでローカルメモリjのどの番地にロードするかは共通領域送信手段1が行う。

【0017】また、各プロセッサで独立な部分e、fは各プロセッサでユーザプログラムを実行する時にスタック領域に書き込まれている必要がある情報であり、スタックの開始番地は各プロセッサで任意に設定できるため、ローカルメモリj、kのどの領域にロードするかは各プロセッサの独立領域書き込み手段5が行う。

【0018】

【発明の効果】本発明の方式によれば、アプリケーションプログラムを各プロセッサで共通な部分と独立な部分に分離し、各プロセッサで共通な部分についてはホストコントローラからは1つのプロセッサにロードすることにより、全てのプロセッサへロードすることができるのでホストと各プロセッサ間の通信データ量を削減できる。

【0019】つまり、通信能力の高いプロセッサ間通信を使用することで、通信処理能力の低いホストとプロセッサ間のデータ量を削減することができるので、プログラム引数を変えるなどの方法で同一のプログラムの別々の条件で実行することも可能であり、さらに各プロセッサへロードする時間を短縮することができるという効果がある。

【図面の簡単な説明】

【図1】本発明の一実施例を示すブロック図である。

【図2】従来の一例を示すブロック図である。

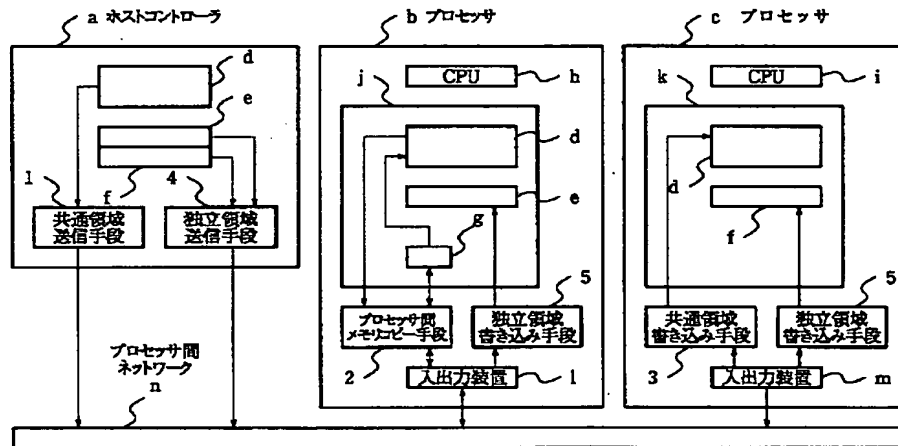
【符号の説明】

- 1 共通領域送信手段
- 2 プロセッサ間メモリコピー手段
- 3 共通領域書き込み手段
- 4 独立領域送信手段
- 5 独立領域書き込み手段
- a ホストコントローラ
- b、c プロセッサ

d アプリケーションプログラムで各プロセッサに共通な部分
 e、a アプリケーションプログラムで各プロセッサに独立な部分
 g 受信データバッファ
 h、i CPU
 j、k ローカルメモリ

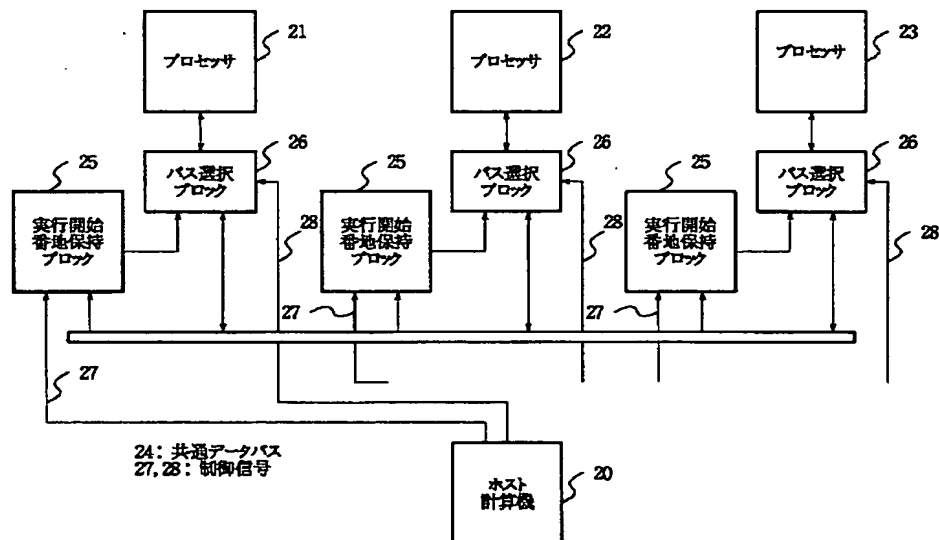
l、m 入出力装置
 n プロセッサ間ネットワーク
 21、22、23 プロセッサ
 24 データバス
 25 実行開始番地情報保持ブロック
 26 バス選択ブロック。

【図 1】



d: アプリケーションプログラムで各プロセッサに共通な部分
 e, f: アプリケーションプログラムで各プロセッサに独立な部分
 g: 受信データバッファ
 j, k: ローカルメモリ

【図 2】



24: 共通データバス
 27, 28: 制御信号